



**HAL**  
open science

# Alignement des points de vue du système d'information Une approche pragmatique

Jonathan Pepin, Pascal Andre, Christian Attiobé, Erwann Breton

## ► To cite this version:

Jonathan Pepin, Pascal Andre, Christian Attiobé, Erwann Breton. Alignement des points de vue du système d'information Une approche pragmatique. Actes du XXXVIème Congrès INFORSID, May 2018, Nantes, France. hal-02468406

**HAL Id: hal-02468406**

**<https://hal.science/hal-02468406>**

Submitted on 5 Feb 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

---

# Alignement des points de vue du système d'information

## Une approche pragmatique

**Jonathan Pepin**<sup>1,2</sup>, **Pascal André**<sup>1</sup>, **Christian Attiogbé**<sup>1</sup>,  
**Erwann Breton**<sup>2</sup>

1. LS2N CNRS UMR 6004 Université de Nantes 2, rue de la Houssinière, BP 92208,  
F-44322 Nantes Cedex 3

*Prenom.Nom@univ-nantes.fr*

2. Mia-Software - Nantes 11, rue Nina Simone 44009 Nantes Cedex 1

*ebreton@sodifrance.fr*

---

*RÉSUMÉ. La maintenance de systèmes d'informations implique de mettre en correspondance la vision stratégique du métier et la vision informatique, parfois via le prisme de l'urbanisation. La distance sémantique entre les points de vue rend difficile la mesure de l'évolution du parc applicatif vis-à-vis des processus métiers ou des technologies. Nous proposons une approche pragmatique pour rapprocher les points de vue et aider à évaluer l'impact de restructurations sur l'évolution du parc applicatif. Une fois alignés les modèles des deux points de vue, des mesures estiment la qualité de l'alignement. L'approche présentée est mise en œuvre par des transformations de modèles et expérimentée sur des cas concrets.*

*ABSTRACT. The maintenance of Information Systems involves to fit the business vision with the Information Technology vision, possibly with the enterprise architecture. The semantic distance between viewpoints makes it difficult to evaluate the impact of the application portfolio evolution with respect to business processes or technologies. We propose a pragmatic approach to align the different viewpoints and help to estimate the impact of restructuring actions on the legacy application portfolio. Once models are aligned from both points of view, we propose measurements that estimate the quality of the alignment. This approach has been implemented by model transformations and experimented on concrete cases.*

*MOTS-CLÉS : Systèmes d'information - Architecture d'entreprise - Rétro-ingénierie - Alignement Métier - Ingénierie des modèles - Mesure*

*KEYWORDS: Information Systems, Enterprise Architecture, Reverse Engineering, Business-IT Alignment, Model Driven Engineering*

---

## 1. Introduction

Dans un système d'information (SI), la cohérence entre l'organisation et le système informatique qui implante ses processus automatisés est fondamentale du point de vue de la qualité du SI. On parle d'*alignement business/IT* lorsque les moyens mis en œuvre (pas seulement l'informatique) sont cohérents avec la stratégie d'entreprise et contribuent efficacement à sa compétitivité (Henderson, Venkatraman, 1993). Cet alignement contribue à la performance de l'organisation (Thevenet, Salinesi, 2007 ; Ullah, Lai, 2013 ; Roelens *et al.*, 2017). Face à la concurrence, les entreprises raccourcissent le cycle de décision et exigent une forte réactivité du SI alors que le cycle de maintenance et de renouvellement du parc applicatif, souvent hétérogène, est plus long du fait de contraintes budgétaires ou organisationnelles. Cet alignement est un défi lancé depuis plus de 20 ans avec l'article fondateur (Henderson, Venkatraman, 1993) et qui reste d'actualité (Coltman *et al.*, 2015). Il a évolué au cours du temps notamment avec l'émergence de l'*architecture d'entreprise* (ou *urbanisation*).

La démarche d'architecture d'entreprise permet de faire évoluer le système d'information d'une entreprise au rythme des stratégies à appliquer pour faire progresser l'activité de l'entreprise. Des plans de transitions (ou plans de migrations) se mettent en place selon un cycle itératif permettant l'évolution de sous-ensembles ciblés du système d'information (Lankhorst, 2013). La cartographie vise à obtenir une image fidèle de l'état actuel des sous-ensembles du système d'information. Elle est constituée de *points de vue* du système d'information, notamment les points de vue des domaines *métier* et *informatique*. Les points de vue ont des objectifs, des méthodes, des acteurs, des représentations ou des pratiques variées. Les points de vue sont complémentaires, couvrant des aspects différents (composition), ou similaires mais avec des niveaux de détail différents (abstraction). Par exemple, il est nécessaire de structurer les différentes sources, support de la cartographie, hétérogènes par nature. D'une part, la stratégie et le fonctionnement de l'entreprise sont renseignés à travers des documents le plus souvent informels et dans le meilleur des cas à travers une modélisation métier décrite dans un langage plus ou moins formalisé et normé. D'autre part, le patrimoine applicatif est composé de programmes et d'infrastructures physiques. Sans cartographie applicative, la documentation est constituée au mieux de modèles d'architecture. Le cas échéant le code source est un point d'entrée possible, mais sa densité nécessite une abstraction méthodique des concepts techniques par détection ou filtre. Il peut être difficile de garder ces points de vue cohérents au fil des **maintenances**, des *changements technologiques* (mobilité, géolocalisation, objets connectés...) et des *évolutions stratégiques et organisationnelles* (lois, marché concurrentiel, actionnariat, économie collaborative, préoccupations environnementales...). L'évolution des points de vue du SI est un problème complexe et un enjeu important pour l'entreprise. Elle permet à l'entreprise d'être à la fois *performante et flexible*. Pour suivre et piloter l'évolution du SI, la démarche d'architecture d'entreprise est un projet qui mobilise un temps conséquent et du personnel en nombre ce qui peut *engendrer des coûts* importants.

Les travaux présentés ici s'inscrivent dans les étapes de cartographie et d'identification de la trajectoire du SI. Aligner de bout en bout chaque élément du SI, du code

binaire déployé jusqu'à la stratégie d'entreprise, reste encore une gageure. Notre objectif est de proposer une *méthode pragmatique d'alignement par les modèles* pour les systèmes d'information existants qui nécessitent des évolutions. L'alignement *business/IT* n'est qu'une étape de la démarche d'architecture d'entreprise et d'urbanisation (Ullah, Lai, 2013) mais elle est cruciale du point de vue de l'évolution du SI. Dans le cycle de la méthode de développement d'architecture (*Architecture Development Method, ADM*) du cadre d'architecture TOGAF (*The Open Group Architecture Framework*) présenté par la FIGURE 1, nos travaux assisteraient à l'accomplissement des étapes notées B, C, D et E. Notre méthode contribue à l'alignement entre les

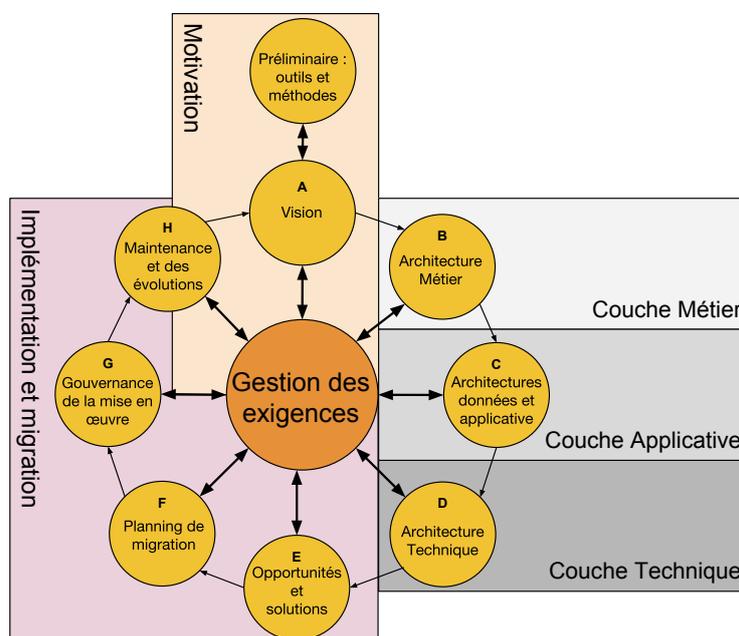


Figure 1. Le cycle ADM du cadre d'architecture d'entreprise TOGAF

visions métier et technique (*business/IT*) pour maintenir le patrimoine applicatif en phase avec l'évolution des métiers. Nous avons exposé la méthode dans (Pepin *et al.*, 2015). Nous en présentons ici l'évaluation et l'extension aux couches hautes et basses de l'architecture d'entreprise.

Cet article est organisé comme suit. Dans la section 2 nous exposons notre approche opérationnelle de l'alignement des visions métier et technique ; cette approche est orientée modèles et le cœur du problème concerne les vues métier, applicative et fonctionnelle. Nous proposons une solution non-intrusive par tissage. L'évaluation de l'alignement est discutée dans la section 3. Notre approche est outillée et a été expérimentée sur plusieurs cas concrets relatés dans la section 4. Le cœur opérationnel est étendu dans sa partie basse à l'infra-structure et dans sa partie haute à la stratégie; nous en discutons la vision dans la section 5. La section 6 résume l'état des travaux et trace des perspectives.

## 2. Une approche pragmatique au cœur de l'alignement

Notre problème est de définir une méthode d'alignement qui s'applique aux systèmes d'information existants (*legacy systems*) qui nécessitent des évolutions. Nous ne considérons que la dimension alignement *business-IT* de (Henderson, Venkatraman, 1993). Dans une vision classique, cet alignement est interprété comme une ligne de traçabilité traversant les couches (partie gauche de la FIGURE 2) mais comme indiqué en introduction aligner de bout en bout reste encore une gageure. L'expérimentation et la pratique des cas industriels nous ont montré que cette vision classique du SI est *idéaliste* : les entreprises n'ont pas toutes le même niveau de **maturité** ni la capacité à s'investir dans un projet d'architecture du SI. Ainsi, certaines entreprises n'abordent que l'essentiel du SI avec les vues applicative et fonctionnelle, tandis que d'autres poussent l'exercice jusqu'aux couches de processus métier et technique. De plus un SI est rarement cartographié en entier, le travail d'architecture commence lorsque surviennent des problèmes : techniques, de maintenance ou de coût. Lorsque les problèmes deviennent majeurs et ne peuvent plus être réparés par l'application de pansement (*Patch*), un audit complet est nécessaire pour redéfinir une vision stratégique du SI par la cartographie de l'existant. Ainsi, l'architecture va s'attaquer à *une ou plusieurs applications* mises en cause dans l'entreprise et non à l'intégralité du SI.

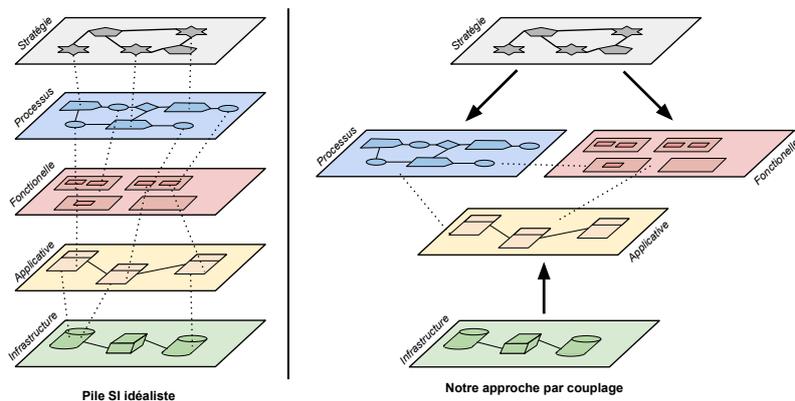


Figure 2. Les couches du SI : approche idéale vs. approche pragmatique

En pratique le but est surtout de rapprocher les modèles et de mettre en évidence les forces et faiblesses pour des scénarios d'évolution. Dans nos travaux précédents (Pepin *et al.*, 2016) nous avons présenté une vision opérationnelle, illustrée à droite sur la FIGURE 2. Le cœur de l'alignement est formé d'un triplet de points de vue : métier, fonctionnel et applicatif. La traçabilité est mise en œuvre par des liens sémantiques variés entre ces points de vue (ou couche). Le point de vue métier décrit les processus métier du SI. Le point de vue application représente les applications du SI sous forme de composants et services logiciels. Le point de vue fonctionnel représente l'urbanisation du système d'information *e.g.* en zone/îlots/quartiers. Selon le niveau de maturité, la préoccupation, ou l'importance de l'entreprise, on disposera

ou pas des points de vue métier et fonctionnel. L'approche est *flexible* (agile) car elle s'adapte aux éléments disponibles. Il sera toujours possible d'ajouter des points de vue à l'alignement. Les liens sémantiques expriment des relations de raffinement ou de correspondance pour les données et les traitements. La FIGURE 3 est un extrait des méta-modèles de ces points de vue et de leur alignement détaillés dans (Pepin, 2016).

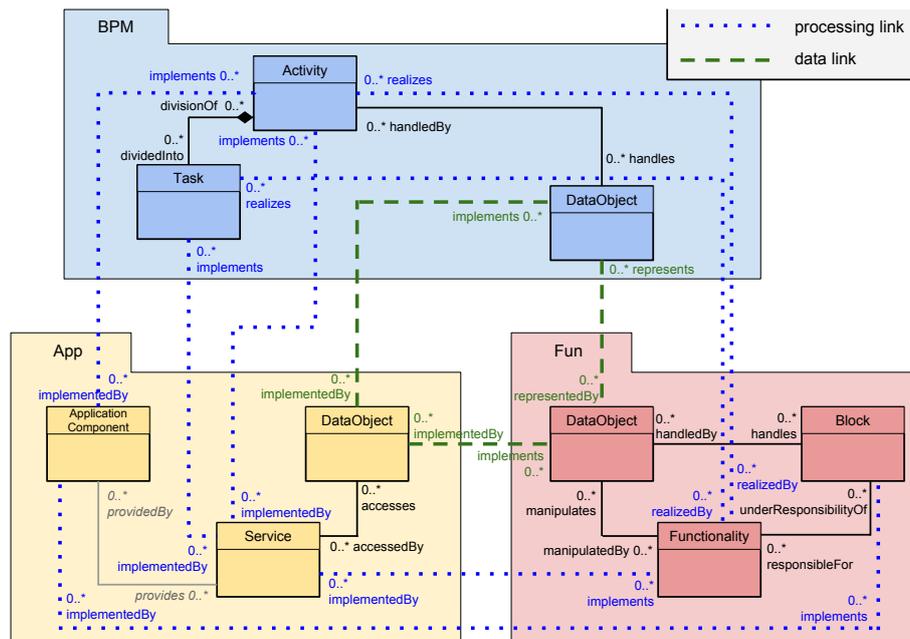


Figure 3. Définition de l'alignement au niveau méta-modèle

Le rapprochement des points de vue se fait en concrétisant la stratégie par des modèles de processus métier (*top-down*) et en masquant les détails d'implantation (*bottom-up*) jusqu'à un niveau acceptable pour un "langage commun". Le langage commun est alors défini comme un tissage entre les langages *i.e.* une correspondance entre leurs concepts. A ce stade, nous faisons abstraction de la vision stratégique, nous y reviendrons en section 5.2. A l'autre bout de la chaîne, le code déployé implique une maîtrise de l'infrastructure pas toujours disponible, nous y reviendrons en section 5.1.

L'alignement s'inscrit dans la première étape du travail de l'architecte dans le cycle ADM (FIGURE 1) : établir la cartographie du SI (B et C). Le processus d'alignement vise à alimenter les modèles *BPM*, *App* et *Fun* en fonction des données disponibles dans l'organisation puis à les aligner concrètement. Pour obtenir le modèle applicatif, nous avons proposé un processus de remontée en *abstraction* depuis le code source des applications (rétro-ingénierie). Une série de transformations permet d'abstraire les concepts techniques pour ne conserver que les éléments architecturaux. Pour obtenir les modèles métiers, c'est l'inverse, on doit matérialiser la stratégie d'entreprise, on parle de *concrétisation* ou de raffinement. Si la documentation métier est inexistante ou partielle, les différents acteurs (analystes métier, analystes fonctionnels et utilisateurs)

teurs finaux) doivent définir les modèles fonctionnels et processus à partir de leurs savoirs et connaissances du fonctionnement et de l'organisation de l'entreprise. Les processus ont été implantés sous Eclipse avec Modisco et Mia-transformation. Les détails sont fournis dans le chapitre 5 de (Pepin, 2016). Pour aligner des modèles obtenus, nous proposons une solution non-intrusive de *tissage* d'implémentation de facettes permettant l'extension virtuelle de méta-modèles. Nos propositions forment une méthode *outillée* présentée dans (Pepin *et al.*, 2018) et qui utilise des standards de l'ingénierie des modèles. Nous avons notamment contribué au framework EMF Facet.

En résumé, nous proposons une méthode qui consiste à outiller la démarche d'architecture d'entreprise dans les étapes de cartographie, d'alignement et de prise de décision à la transformation du SI. Notre méthode est **générique** et **compatible** avec les principaux cadres d'architecture abordés précédemment. Nous avons vu qu'il n'est pas nécessaire d'adopter un cadre unique, mais qu'il est possible, voire recommandé de prendre les **bonnes pratiques** adaptées à la situation dans différents cadres. Nos travaux n'ont aucune adhérence forte avec l'un des cadres d'architecture d'entreprise. Les modèles sont construits et alignés pour être analysés et répondre aux interrogations des décideurs. Nous y répondons dans la section 3.

### 3. Maîtriser les évolutions du SI par analyse de dépendances

La cartographie et l'alignement des points de vue sont les étapes préliminaires d'un processus d'évolution du SI tel que celui de la FIGURE 1. L'architecte d'entreprise doit être capable d'identifier les éléments du SI à faire évoluer et de mesurer les impacts sur les différentes couches (ou points de vue). Les points de vue étant transverses (partie droite de la FIGURE 2), l'alignement que nous avons proposé à partir de liens sémantiques permet à l'architecte de *d'interroger le modèle i.e.* quels composants applicatifs implémentent telle fonctionnalité ? Quels processus métiers sont impactés par des modifications de composants applicatifs ou techniques ? L'objectif est de fournir aux urbanistes des outils d'assistance à l'évolution.

L'évaluation de l'alignement vise à détecter des problèmes de cohérences ou de complétude entre couches. Nous avons dans un premier temps travaillé sur une mesure de la qualité de l'alignement avec un indicateur global permettant d'étalonner la cohérence globale du système d'information. Nous n'avons pas trouvé de calcul d'un agrégat pertinent. L'analyse détaillée nous a convaincu que ce n'était ni la voie, ni le besoin. Ce n'est pas la voie car trouver un référentiel de bonne qualité de l'alignement se révèle ardu et finalement trop complexe. Ce n'est pas le besoin car l'objectif des architectes n'est pas d'aboutir après plusieurs itérations à un "super" alignement final, mais plutôt d'établir un constat à un moment donné avec des pistes d'améliorations puisque le système d'information évolue en permanence. L'évaluation peut se faire par observation/modification, par une analyse structurelle ou par des requêtes spécifiques. Dans la suite nous présentons trois techniques : l'outil de tissage, l'analyse par requêtes et l'analyse de dépendances.

### 3.1. Observer et modifier l'alignement du SI par tissage

L'alignement de modèles réalisé à l'aide de l'approche de la section 2 permet une navigation bi-directionnelle, *montante et descendante* entre les points de vue du SI. Un éditeur arborescent compatible avec le framework Eclipse EMF permet de charger les modèles et les liens sémantiques d'alignement permettent la navigation. Nous proposons un éditeur de tissage pour créer, consulter, modifier ou supprimer les liens d'alignement entre les trois points de vue exhibés dans la section 2. Cet éditeur est donc un navigateur dans l'alignement. La FIGURE 4) illustre cet éditeur sur un cas de simulation bancaire. A droite, le point de vue applicatif comprend différents services et fonctions et le point de vue fonctionnel met en évidence le bloc Adhérent d'un contrat IARD. A gauche le tissage relie le bloc fonctionnel *Simulation contrat* avec les services du modèle applicatif et notamment le service *validerSimulationContrat*.

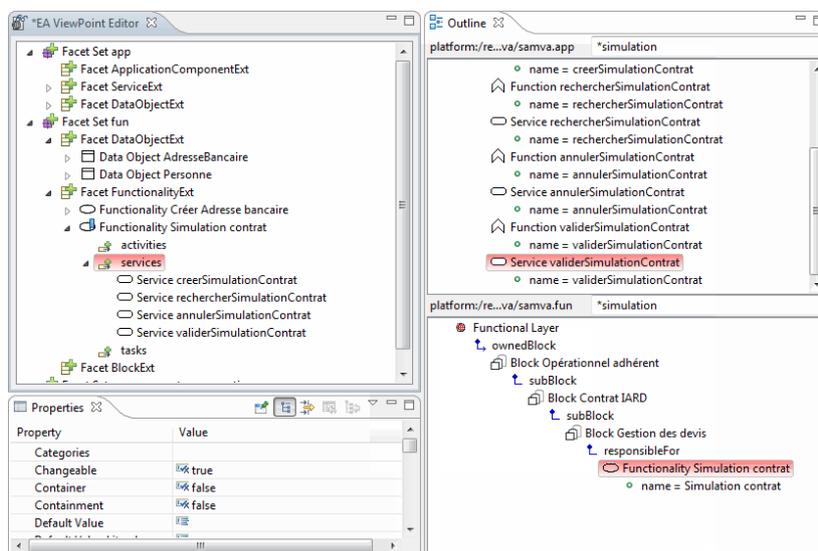


Figure 4. Editeur de tissage pour la navigation entre modèles

Naviguer et modifier le tissage ne suffit pas pour évaluer. D'une part, la navigation arborescente même assistée par des filtres de recherche est peu pratique pour les modèles volumineux. D'autre part, elle ne répond qu'aux seuls types de questions posées précédemment. Nous souhaitons en poser d'autres comme "quels sont les éléments qui ne sont pas alignés correctement?". . . Afin de pallier ce problème, nous proposons deux types d'analyses dans les sections suivantes.

### 3.2. Mesurer l'alignement du SI par requêtes

Le travail d'alignement est proportionnel au volume d'informations cartographiant les points de vue du SI. L'architecte doit pouvoir suivre son avancement et vérifier sa

cohérence, mais aussi mesurer l'alignement pour fournir aux décideurs des indicateurs guidant à la fois l'analyse de l'état actuel, la détection d'anomalies et la valorisation de scénario de projections futures.

La mesure de l'alignement reste inexplorée (Aversano *et al.*, 2016). Les travaux de Simonin (Simonin, 2009) se focalisent sur la couche fonctionnelle (urbanisation). D'autres traitent surtout de la couche métier (voire stratégique) avec le logiciel. Ils incluent la notion d'acteurs, qui n'a pas forcément de sens du point de vue logiciel. Aversano *et al.* (Aversano *et al.*, 2010) traitent de la couverture des processus métiers par le logiciel, mais ne présentent pas les modèles du logiciel. Ils ont proposé des métriques pour l'*alignement fonctionnel* dans (Aversano *et al.*, 2016) mais pour des modèles de bas niveaux (activités et classes UML). Nous avons défini des métriques d'indice de couplage et de cohérence similaires mais leur intérêt restait difficile à appréhender car l'alignement est multiforme. Rolland et Etien (Etien, Rolland, 2005) alignent les processus métiers avec des modèles UML via des ontologies mais sans parc applicatif. Thévenet *et al.* (Thevenet, Salinesi, 2007) s'intéressent à l'alignement stratégique et l'évolution, sans lien direct avec le code.

Nos indicateurs sont classés en quatre catégories : couverture, cohérence, densité d'alignement concret (entre les modèles métiers, applicatifs et fonctionnels) et couverture du code final. Le *framework* Eclipse EMF permet d'interroger les modèles par requêtes écrites dans le langage OCL. Nous avons étendu OCL pour que les requêtes soient compatibles avec EMF Facet, utilisé pour le tissage entre modèles. Les requêtes servent à mesurer les indicateurs d'alignement. Ainsi, une première mesure possible est la complétude de l'alignement illustré par la FIGURE 5 : le nombre d'éléments alignés sur le nom d'éléments alignables des points de vue processus et applicatif, selon la définition de l'alignement (FIGURE 3) entre points de vue au niveau méta-modèle.

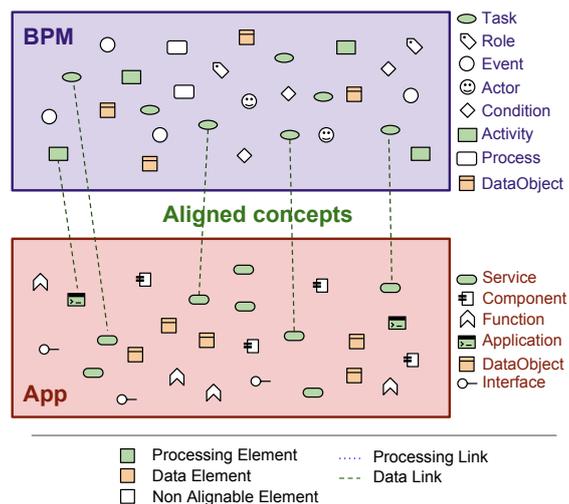


Figure 5. Complétude Activités de Processus/Service applicatif

Les concepts sont typés ce qui autorise des analyses par préoccupation (*concerns*) e.g. données et traitements sont distingués. Les requêtes permettent la vérification de cohérence entre traitements et données. Si un élément de traitement d'un point de vue A qui manipule des éléments de données, est aligné avec un élément de traitement d'un autre point de vue B, alors les éléments de données du point de vue A doivent être alignés aux éléments de données du point de vue B et réciproquement. La requête 1 liste les blocs ayant un alignement incohérent de données avec des composants applicatifs.

Listing 1 – Oublis entre Bloc Fonctionnels et Composants applicatifs

---

```
BlockAppComponentNonConsistent:
context FonctionnalLayer
fun::Block.allInstances()->select(applicationComponents->notEmpty())->
  symmetricDifference(
fun::Block.allInstances()->select(blk | blk.applicationComponents.
  provides.accesses.dataObjectsFun.handles->includes(blk)))
```

---

L'analyse par requête de l'alignement détecte des oublis et incohérences et de les corriger à chaque itération permettant un réel suivi qualité en continu du SI. Des règles d'urbanisme spécifiques au SI peuvent aisément être implémentées en OCL et compatibles avec les liens d'alignement.

### 3.3. Identifier les dépendances entre points de vue

Un SI doit avoir des qualités de cohérence forte et couplage faible, bien connues en conception logicielle (Parnas, 1972). Le SI au fil des évolutions peut devenir intriqué. L'analyse de dépendances répond ainsi à des problématiques très diverses. Du point de vue applicatif, elle permet à l'architecte de mettre en évidence (i) les zones de couplage fort candidates à une restructuration (*refactoring*) (ii) les impacts en cas de débranchement ou de remplacement de composants applicatifs. Du point de vue processus métiers, elle permet d'identifier les dépendances entre activités mais aussi déterminer les composants applicatifs impactés par une réorganisation métier. Outre le calcul des dépendances, l'architecte a besoin de représentations visuelles des dépendances (à grande échelle) pour se projeter dans l'évolution.

Techniquement, les modèles alignés représentent un graphe composé de nœuds (les instances des types) et d'arcs (les relations entre instances). Ce qui permet d'appliquer des algorithmes issus de la théorie des graphes. Les graphes peuvent se représenter par une matrice, et plus précisément une matrice d'adjacence. Les nœuds sont représentés en-tête de chaque ligne (*l*) et colonne (*c*), une intersection entre une ligne et une colonne est la présence d'une dépendance entre deux nœuds. Cette matrice est appelé *Matrice de structure de dépendance* (DSM) (Eppinger, Browning, 2012). Il existe donc  $l \times c$  possibilités de représenter le graphe en matrice, selon l'ordre de lecture de chaque nœuds. De ce constat quantitatif, nous nous sommes posés la question du meilleur ordre pour représenter notre graphe d'alignement et ainsi de faciliter la lecture de l'architecte pour visualiser les dépendances et notamment identifier les groupes de dépendance. Nous avons étudié les algorithmes de regroupements (*clus-*



de vue applicatif du même exemple (jouet) que la FIGURE 4. Les lignes et colonnes représentent les concepts du modèle applicatif (en jaune) ou fonctionnel (en rose). La ligne diagonale montre que chaque concept est lié à lui-même. Les points intéressants sont les sous-matrices plus pleines comme celui de la ligne 20 et suivantes qui met en évidence un cluster de services, concernant ici les sollicitations clients. En ligne 41-43 on voit un couplage applicatif-fonctionnel sur la gestion des photos. Noter que nos algorithmes de regroupement sont déterministes et fournissent donc un résultat identique quelle que soit l'exécution d'un jeu de valeurs. Noter aussi qu'il s'agit d'un prototype que nous avons développé et que de nombreux points sont à améliorer.

#### 4. Expérimentations

Dans un premier temps, nous avons testé la méthode (principes, démarche, outils) sur des cas simples et imaginaires, comme celui qui a servi à illustrer nos propos dans les sections précédentes. Nous avons eu ensuite l'opportunité de tester notre démarche sur trois cas d'études concrets, provenant de sociétés d'Assurance Mutuelle françaises que nous nommerons SAMM, SAMI et SAMUT. Chaque étude possède ses spécificités et couvre tout ou partie des modèles du processus de la section 2. Il ne s'agissait pas de réaliser un cas de bout en bout et le tissage reste partiel. L'expérimentation a pour but de vérifier la faisabilité de notre démarche : pertinence des modèles vis-à-vis de la pratique en entreprise, adéquation de la démarche, automatisation des transformations et de leur enchaînement, pertinence du tissage et de sa mise en œuvre, application à des modèles volumineux. Nous résumons ci-après les expériences.

Le cas SAMM est composé *i*) d'un code source complet écrit en Java avec 33 400 classes (3 400 000 lignes de codes) et *ii*) d'un référentiel d'entreprise sous la forme d'un portail HTML exporté depuis le logiciel MEGA EA <sup>1</sup>. Le but de ce scénario était de valider que notre méthode permet d'exécuter l'analyse par clustering d'un alignement réel à l'aide de notre matrice de dépendance. Le référentiel d'entreprise contient 360 diagrammes de processus métier couvrant la totalité du SI. Le volume de l'application est conséquent et représente un véritable défi à traiter. Ce cas d'étude nous a permis de tester notre approche sur un source code important, le chargement des modèles obtenus par rétro-ingénierie a été un défi pour nos outils. La transformation a été facilitée par la présence d'une nomenclature des concepts qui se retrouve à différents niveaux et par là même montre de bonnes pratiques de codage, à quelques exceptions près. Néanmoins, nous regrettons ne pas avoir eu accès au source original du référentiel MEGA pour réaliser un tissage complet.

Le cas SAMI est composé uniquement d'un référentiel MEGA dont le source est disponible. Le but de ce scénario était de valider que notre méthode permettait bien d'exécuter une remontée en abstraction d'un code applicatif réel et conséquent. Nous avons extrait les différents concepts pour peupler à l'aide d'une transformation nos différents modèles (App, BPM, Fun) afin de vérifier la couverture et la compatibilité

1. <http://www.mega.com/fr/solution/business-architecture>

des concepts. Le cas présente 625 composants, 11894 services, 18 blocs fonctionnels, 131 fonctionnalités, 167 processus et 268 activités.

Le cas SAMUT n'avait aucune représentation métier. Le but de ce scénario était de valider que notre méthode permet de construire un alignement à partir de zéro en constituant les modèles et d'aider l'architecte d'entreprise à visualiser les dépendances. Les objets de données ont été extraits par rétro-ingénierie d'une base de données hiérarchique et les composants applicatifs de procédures stockées. Un architecte à ensuite modélisé et identifié des blocs fonctionnels et nous avons alors pu réaliser le tissage entre les blocs et les composants applicatifs. Le SI comporte 12 blocs, 1045 composants et 669 objets de données. Ce cas d'étude a permis de tester notre méthode de tissage, et isoler des composants qui étaient orphelins (rangés dans aucun bloc).

Ces trois cas ont des propriétés particulières, les supports sources sont hétérogènes et représentatifs de la disparité de maturité des SI. Ils ont mis en évidence la souplesse de notre méthode qui peut s'adapter à chaque étape. La contrepartie est d'enrichir l'écosystème des modèles et transformations pour chaque nouveau cas rencontré.

## 5. Extensions de l'alignement opérationnel

Dans les sections précédentes, nous nous sommes attachés à réduire le fossé entre domaine métier et informatique. Nous discutons ici de l'extension de l'alignement aux couches infrastructure et stratégie en vue de couvrir l'ensemble des vues du SI.

### 5.1. Alignement d'infrastructure

Dans nos cas d'études, nous avons constaté que les entreprises maintenaient une cartographie technologique composée des informations sur les serveurs, systèmes, réseaux et caractéristiques physiques. Cette cartographie permet à la direction des systèmes d'information (DSI) de réaliser une intervention de maintenance ciblée. Par exemple, en cas de panne d'un service applicatif, il est nécessaire de retrouver rapidement les coordonnées du logiciel incriminé : IP, droits d'accès, site physique ou cloud, etc. Conserver une cartographie technologique actualisée est une forte préoccupation. Une panne entraînant la cessation d'une partie de l'activité de l'entreprise ayant un coût lié au temps d'immobilisation, la résolution doit-être la plus rapide possible pour limiter les pertes. Nous avons établi un premier état de l'art qui met en évidence que certaines cartographies technologiques mélangent tout les aspects, tandis que d'autres distinguent les concepts de déploiement et des concepts d'infrastructure.

**Déploiement** ce point de vue a pour but de décrire comment les applications sont réparties au niveau virtuel. La vue applicative décrit les composants, fonctions ou objets de donnée, mais pas leur nature technologique. Par exemple, un ou plusieurs composants écrits en Java peuvent être encapsulés dans des archives *jar*, *war* ou *ear* déployés dans un serveur d'application J2EE (Glassfish, JBoss, Apache Tomcat...); un ou plusieurs objets de données sont stockés dans une table de base de données (MySQL, Postgre, Oracle, Neo4j, MongoDB...).



la gestion et de l'informatique. Nous restreignons notre domaine d'investigation aux approches incluant des modèles, notamment autour de l'architecture d'entreprise. Le problème se résume alors à disposer de modèles pour la couche stratégique et la couche processus métiers et de modèles pour l'alignement entre ces couches conformément à l'approche de la FIGURE 2. Nous disposons déjà de langages pour la couche métier.

L'étude bibliographique met en évidence les éléments suivants :

- Il existe de nombreux langages et techniques de modélisation de la stratégie ou de maturité (Ullah, Lai, 2011 ; Aversano *et al.*, 2016 ; Roelens *et al.*, 2017), ce qui ne facilite pas la mise en pratique pour les entreprises et la conception d'outils d'aide.

- Certaines approches s'appliquent à la conception (*design time*) (Thevenet, Salinesi, 2007) en établissant des ponts entre couches (traçabilité), l'ingénierie des exigences y joue un rôle prépondérant (Thevenet, Salinesi, 2007 ; Ullah, Lai, 2011). D'autres visent la rétro-conception en annotant les modèles métiers d'indicateurs exploités pour relier à des concepts de la couche stratégique.

- La couche stratégique peut elle-même être décomposée en plusieurs couches notamment en séparant par exemple une approche basée sur les buts (*goal modeling languages* (Ullah, Lai, 2011 ; Doumi *et al.*, 2011), i\* (Pijpers *et al.*, 2008)) et celle basée sur la valeur (VMDL (Roelens, Poels, 2013), e<sup>3</sup>-value (Pijpers *et al.*, 2008)). Une approche telle que *Process Goal Alignment* (PGA) permet de relier les deux niveaux (Roelens *et al.*, 2017).

- La jonction entre les couches métier et stratégique peut se faire en annotant les processus métiers d'informations utiles à l'alignement avec la stratégie. Morrison *et al.* proposent ainsi un calcul d'alignement sur les buts basés sur ces annotations (Morrison *et al.*, 2012). Ullah et Lai proposent de relier les processus métiers aux buts (Ullah, Lai, 2011).

- Plus récemment, le *Business Motivation Model* (BMM) de l'OMG contient des éléments pour spécifier la stratégie et d'autres éléments pour relier le niveau stratégie au niveau métier (Bhattacharya, 2017 ; Hinkelmann, Pasquini, 2014).

Ne disposant pas de modèles de stratégie pour nos études de cas, nous proposons dans un premier temps un système de pondération des activités des processus métiers pour évaluer l'impact des changements sur le système futur. Ces pondérations servent à la mesure d'impact des différents scénarios d'évolution du SI. Par la suite, nous envisageons une mise en œuvre d'une version générique de BMM qui a l'avantage de s'intégrer avec nos standards de modélisation que ce soit TOGAF, UML ou EMF.

## 6. Conclusion

Nous avons proposé une méthode pragmatique au problème d'alignement des points de vue métier et applicatif s'insérant dans la démarche d'urbanisation des architectures d'entreprise. Notre méthode est basée sur une proposition de modèles intermédiaires génériques, un rapprochement des points de vue et un alignement par tissage de concepts comparables. Le rapprochement est rendu possible par une abs-

traction progressive du code en architecture applicative à base de composants et services. Notre approche est outillée dans le cadre d'Eclipse EMF et a été expérimentée sur des cas réels d'entreprise, permettant d'éprouver la viabilité de notre approche. L'application à un source code de taille importante fut un défi mais les modèles obtenus par rétro-ingénierie ont pu être chargés par nos outils.

Certains points sont encore à améliorer dans l'automatisation du processus comme la découverte des composants logiciels ou la détection des correspondances du tissage. Le travail en cours sur les métriques doit permettre la mise en place de tableaux de bord pour l'évaluation du tissage et les scénarios d'évolution. Une autre perspective est l'enrichissement du tissage avec plus de concepts pour mieux prendre en compte les points d'évolution du système d'information, pas seulement la structure (Saat *et al.*, 2010). En particulier, nous souhaitons pouvoir représenter le couplage entre parties de modèles. Dans cette lignée, nous devons mettre à l'épreuve nos méta-modèles vis-à-vis de pratiques (non automatisées) d'urbanisation et d'alignement.

## Bibliographie

- Aversano L., Grasso C., Tortorella M. (2010). Measuring the alignment between business processes and software systems: A case study. In *Proceedings of the 2010 acm symposium on applied computing*, p. 2330–2336. New York, NY, USA, ACM.
- Aversano L., Grasso C., Tortorella M. (2016). Managing the alignment between business processes and software systems. *Information and Software Technology*, vol. 72, p. 171 - 188.
- Bartolomei J., Cokus M., Dahlgren J., Neufville R. de, Maldonado D., Wilds J. (2010). Analysis and applications of design structure matrix, domain mapping matrix, and engineering system matrix frameworks. Consulté sur [http://ardent.mit.edu/real\\_options/Real\\_opts\\_papers/Jennifer%20mini%20thesis.pdf](http://ardent.mit.edu/real_options/Real_opts_papers/Jennifer%20mini%20thesis.pdf)
- Bhattacharya P. (2017). Modelling Strategic Alignment of Business and IT through Enterprise Architecture: Augmenting Archimate with BMM. *Procedia Computer Science*, vol. 121, p. 80 - 88. (CENTERIS 2017)
- Coltman T., Tallon P., Sharma R., Queiroz M. (2015, 01 Jun). Strategic it alignment: twenty-five years on. *Journal of Information Technology*, vol. 30, n° 2, p. 91–100.
- Dongen S. van. (2000). *Graph Clustering by Flow Simulation*. Phd thesis, University of Utrecht.
- Doumi K., Bařna S., Bařna K. (2011). Modeling approach for business IT alignment. In *Proceedings of ICEIS 2011, volume 4, beijing, china*, p. 457–464.
- Eppinger S. D., Browning T. R. (2012). *Design Structure Matrix Methods and Applications*. Cambridge, Mass, MIT Press.
- Etien A., Rolland C. (2005). Measuring the fitness relationship. *Requirements Engineering*, vol. 10, n° 3, p. 184-197. Consulté sur <http://dx.doi.org/10.1007/s00766-005-0003-8>
- Henderson J. C., Venkatraman N. (1993, janvier). Strategic alignment: Leveraging information technology for transforming organizations. *IBM Syst. J.*, vol. 32, n° 1, p. 4–16.

- Hinkelmann K., Pasquini A. (2014). Supporting business and IT alignment by modeling business and IT strategy and its relations to enterprise architecture. In *Enterprise systems conference, ES 2014, shanghai, china, august 2-3, 2014*, p. 149–154. IEEE.
- Lankhorst M. M. (2013). *Enterprise architecture at work - modelling, communication and analysis (3. ed.)*. Springer.
- Morrison E. D., Ghose A. K., Dam H. K., Hinge K. G., Hoesch-Klohe K. (2012). Strategic alignment of business processes. In G. Pallis *et al.* (Eds.), *Service-oriented computing - icsoc 2011 workshops*, p. 9–21. Berlin, Heidelberg, Springer Berlin Heidelberg.
- Parnas D. L. (1972, décembre). On the criteria to be used in decomposing systems into modules. *Commun. ACM*, vol. 15, n° 12, p. 1053–1058. Consulté sur <http://doi.acm.org/10.1145/361598.361623>
- Pepin J. (2016). *Architecture d'entreprise : alignement des cartographies métiers et applicatives du système d'information*. Thèse, Université Bretagne Loire.
- Pepin J., André P., Attiogbé C., Breton E. (2016). Using ontologies for enterprise architecture integration and analysis. *CSIMQ*, vol. 9.
- Pepin J., André P., Attiogbé C., Breton E. (2018). Virtual extension of meta-models with facet tools. In *Proceedings of the 6th international conference on model-driven engineering and software development - volume 1: Modelsward*, p. 59-70. SciTePress.
- Pepin J., André P., Attiogbé C., Breton E. (2015). A method for business-it alignment of legacy systems. In *Proceedings of ICEIS 2015, volume 3, barcelona, spain, 27-30 april, 2015*.
- Pijpers V., Gordijn J., Akkermans H. (2008). Business strategy-it alignment in a multi-actor setting: A mobile e-service case. In D. Fensel, H. Werthner (Eds.), *Proceedings of the 10th international conference on electronic commerce 2008*, vol. 342. ACM.
- Roelens B., Poels G. (2013). Towards an integrative component framework for business models: Identifying the common elements between the current business model views. In *Proceedings of the caise'13 forum*, vol. 998, p. 114–121. Valencia, Spain, CEUR-WS.org.
- Roelens B., Steenacker W., Poels G. (2017, 13 Jan). Realizing strategic fit within the business architecture: the design of a process-goal alignment modeling and analysis technique. *Software & Systems Modeling*.
- Saat J., Franke U., Lagerstrom R., Ekstedt M. (2010). Enterprise architecture meta models for IT/Business alignment situations. In *Enterprise distributed object computing conference (EDOC), 2010 14th IEEE international*, p. 14–23.
- Schaeffer S. E. (2007). Graph clustering. *Computer science review*, vol. 1, n° 1, p. 27–64.
- Simonin J. (2009). *Conception de l'architecture d'un système dirigée par un modèle d'urbanisme fonctionnel*. Thèse de doctorat non publiée, Université de Rennes 1.
- Thevenet L., Salinesi C. (2007). Aligning IS to organization's strategy: The instal method. In *Advanced information systems engineering, 19th international conference, caise 2007, trondheim, norway, june 11-15, 2007, proceedings*, p. 203–217.
- Ullah A., Lai R. (2011). Modeling business goal for business/it alignment using requirements engineering. *Journal of Computer Information Systems*, vol. 51, n° 3, p. 21-28.
- Ullah A., Lai R. (2013, avril). A systematic review of business and information technology alignment. *ACM Trans. Manage. Inf. Syst.*, vol. 4, n° 1, p. 4:1–4:30.